A Survey Cost Algorithms for Reducing High Toll Transactions

[1]Vinay Singh, [2]Dr. S.Gavaskar
[1]Dept. of CSE, Final Year, Galgotias University, Greater Noida, India
[2]Dept. of Computer Science, Galgotias University, Greater Noida, India

Abstract

Mining Cost Efficient item-sets from a transactional database refers to the discovery of transaction sets with Cost Efficient characteristics improving all profits. Although a number of important algorithms have been proposed in recent years, they have the problem of producing a large number of candidate item-sets for Cost Efficient item-sets. Such a large number of candidate item-sets degrade the mining performance in terms of execution time and space requirement. The condition may become worse when the database contains lots of long transactions or long Cost Efficient item-sets. In this paper, we propose new algorithm, for mining Cost Efficient item-sets with a set of effective strategies for pruning candidate item-sets. The information of Cost Efficient item-sets is maintained in a tree-based data structure such that candidate item-sets can be generated efficiently with only two scans of database.

Keywords

Data Mining, High Utility Item Sets Transactional Databases

I. Introduction

Data mining is an disciplinary field, the confluence of a set of disciplines, including systems based on data base, statistics, machine learning, visualization, and information science .Moreover, depending on the data mining approach to be used, techniques from other disciplines may be applied, such as neural networks, fuzzy and rough set theory, knowledge representation, inductive programming, high-performance computing. Depending on the kinds of data mined on the given data mining application, the data mining system may also represent techniques from spatial data analysis, information retrieval, recognition of any pattern, analysis of image, signal processing, simulation, Web technology, business, bioinformatics, or psychology. Because of the various of disciplines contributing to datamining, data mining research is expected to generate a large variety of data mining systems. Therefore, it is necessary to provide a proper arrangement of data mining systems, which may help potential users differentiate between such systems and identify those that match their needs. Data mining systems can be mentioned according to various criteria, as follows:

A. Clustering

Clustering is the task of partition the points into expected groups called clusters, such that points within a group are very similar, whereas points across clusters are as different hierarchical, density-based, graph-based and spectral clustering. It starts with representative-based clustering methods which include the K-means and Expectation-Maximization (EM) algorithms. K-means is a greedy algorithm that minimizes the squared error of points from their respective cluster means, and it performs hard clustering, that is, each point is assigned to only one cluster. Classification The classification task is to predict the label or class for a given unlabeled point. Formally, a classifier is a model or function M that predicts the class label ˆy for a given input example x, that is, ˆy = M(x), where ˆy ? {c1, c2, · · · ,ck} is the predicted class label (a categorical attribute value). To build the model we require a set of points with their correct class labels, which is called a training set.

B. Classification

The classification task is to predict the label or class for a given unlabeled point. Formally, a classifier is a model function M that predicts the class label ˆy for a given input example x, that is, ˆy = M(x), where ˆy ? {c1, c2, · · · ,ck} is the predicted class label (a categorical attribute value). to be estimated which scales as O(d2). The naive Bayes classifier makes the simplifying assumption that all attributes are independent, which requires the estimation of only O(d) we require a set of points with their correct class labels, which is called a training set.the join) parameters. It is, however, surprisingly effective for many datasets. In this topic we consider the popular decision tree classifier, one of whose strengths is that it yields models that are easier to understand compared to other methods.

The Support Vector Machine (SVM) is one of the most useful classifiers for many different problem domains. The goal of SVMs is to find the most advantageous hyperplane that maximizes the margin between the classes. Via the kernel trick SVMs can be used to find non-linear boundaries, which nevertheless correspond to some linear hyperplane in some high-dimensional “non-linear” space. One of the important tasks in classification is to assess how good the models are.

II. Transactional Data Mining

Association Rule Mining (ARM) identifies frequent itemsets from databases and generates association rules by considering each item in identical value. However, items are actually dissimilar in many aspects in a number of real application, such as retail marketing, networking log, etc. The difference between items makes a strong impact on the decision making in these applications. Therefore, conventional ARM cannot meet the need arising from these application. By considering the different values of unique items as utility, utility mining focuses on identifying the itemsets with high utilities. As “downward closure property” doesn’t apply to utility mining, the generation of candidate itemsets is the most costly in terms of time and memory space.

A. Issues in Transaction Mining

The scope of issues in data mining regarding mining methodology, user interaction, performance, and diverse data types. These issues are introduced below:

B. Mining Methodology and User Interaction Issues

These reflect the kinds of knowledge mined, the ability to mine knowledge at multiple granularity, the use of domain knowledge, improvised mining, and awareness visualization. Mining different kinds of data in databases: Because different users can be involved in different kinds of knowledge, data mining should cover a wide range of data analysis and knowledge discovery task, including data characterization, association and correlation analysis, classification, prediction, clustering, outlier analysis, and evolution analysis (which includes trend and similarity analysis).

C. Incorporation of Background Knowledge

Background knowledge, or information regarding the domain under study, may be used to guide the discovery process and allow discovered patterns to be expressed in concise terms and at different levels of abstraction. Domain knowledge related to databases, such as integrity constraints and deduction rules, can help focus and speed up a data mining process, or judge the interestingness of discovered patterns.

D. Data Mining Query Languages and Ad-Hoc Data Mining

Relational query languages (such as SQL) allow users to pose ad hoc queries for data retrieval. In a similar vein, high-level data mining query languages need to be developed to allow users to describe ad hoc data mining tasks by facilitating the specification of the relevant sets of data for analysis, the domain knowledge, the kinds of knowledge to be mined, and the conditions and constraint to be compulsory on the revealed patterns. Such a language should be included with a database or data warehouse query language and optimized for efficient and flexible data mining.

E. Presentation and Visualization of Data Mining Results

Discovered knowledge should be expressed in high-level languages, visual representations, or other communicative forms so that the knowledge can be easily understood and openly usable by humans. This is mainly crucial if the data mining system is to be interactive. This requires the system to approve expressive knowledge depiction techniques, such as trees, tables, rules, graphs, charts, crosstabs, matrices, or curves.

F. Handling Noisy or Incomplete Data

The data stored in a database may reflect noise, exceptional cases, or incomplete data objects. When mining data regularities, these objects may confuse the process, causing the knowledge model constructed to over fit the data. As a result, the accuracy of the discovered patterns can be poor. Data cleaning methods and data analysis methods that can handle noise are required, as well as outlier mining methods for the discovery and analysis of exceptional cases.

G. Pattern Evaluation—The Interestingness Problem

A data mining system can uncover thousands of patterns. Many of the patterns discovered may be uninteresting to the given user, either because they represent common knowledge or lack novelty. Several challenges remain regarding the development of techniques to assess the interestingness.

III. Performance Issues

A. Efficiency and Scalability of Data Mining Algorithms

To effectively extract information from a huge amount of data in databases, data mining algorithms must be efficient and scalable. In other words, the running time of a data mining algorithm must be predictable and acceptable in large databases. From a database perspective on knowledge discovery, efficiency and scalability are key issues in the implementation of data mining systems. Many of the issues discussed above under mining methodology and user interaction must also consider efficiency and scalability.

B. Parallel, Distributed, and Incremental Mining Algorithms

The huge size of many databases, the wide distribution of data, and the computational complexity of some data mining methods are factors motivating the development of parallel and distributed data mining algorithms. Such algorithms divide the data into partitions, which are processed in parallel. The results from the partitions are then merged.Moreover, the high cost of some data mining processes promotes the need for incremental data mining algorithms that incorporate database updates without having to mine the entire data again “fromscratch.” Such algorithms perform knowledge modification incrementally to amend and strengthen what was previously discovered.

C. Issues Relating to the Diversity of Database Types

1. Handling of Relational and Complex Types of Data

Because relational databases and data warehouses are widely used, the development of efficient and effective data mining systems for such data is important. However, other databases may contain complex data objects, hypertext and multimedia data, spatial data, temporal data, or transaction data. It is unrealistic to expect one system to mine all kinds of data, given the diversity of data types and different goals of data mining. Specific data mining systems should be constructed for mining specific kinds of data. Therefore, one may expect to have different data mining systems for different kinds of data.

2. Mining Information from Heterogeneous Databases and Global Information Systems

Local- and wide-area computer networks (such as the Internet) connect many sources of data, forming huge, distributed, and heterogeneous databases. The discovery of knowledge from different sources of structured, semi-structured, or unstructured data with diverse data semantics poses great challenges to data mining. Data mining may help disclose high-level data regularities in multiple heterogeneous databases that are unlikely to be discovered by simple query systems and may improve information exchange and interoperability in heterogeneous databases. Web mining, which uncovers interesting knowledge about Web contents, Web structures, Web usage, and Web dynamics, becomes a very challenging and fast-evolving field in data mining.

IV. Related Work

Liu, Mengchi, et al[1]. High utility itemsets refer to the sets of items with high utility like profit in a database, and efficient mining of high utility itemsets plays a crucial role in many reallife applications and is an important research issue in data mining area. To identify high utility itemsets, most existing algorithms first generate candidate itemsets by overestimating their utilities, and subsequently compute the exact utilities of these candidates. These algorithms incur the problem that a very large number of candidates are generated, but most of the candidates are found out to be not high utility after their exact utilities are computed. In this paper, we propose an algorithm, called HUI-Miner (High Utility Itemset Miner), for high utility itemset mining. HUI-Miner uses a novel structure, called utility-list, to store both the utility information about an itemset and the heuristic information for pruning the search space of HUI-Miner. By avoiding the costly generation and utility computation of numerous candidate itemsets, HUI-Miner can efficiently mine high utility itemsets from the utilitylists constructed from a mined database.

Samadi, Pedram,et al [2] In this paper, we consider a smart power infrastructure, where several subscribers share a common energy source. Each subscriber is equipped with an Energy Consumption Controller (ECC) unit as part of its smart meter. Each smart meter is connected to not only the power grid but also a communication infrastructure such as a local area network. This allows two-way communication among smart meters. Considering the importance of energy pricing as an essential tool to develop efficient demand side management strategies, we propose a novel real-time pricing algorithm for the future smart grid. We focus on the interactions between the smart meters and the energy provider through the exchange of control messages which contain subscribers' energy consumption and the real-time price in sequence.

Liu, Ying et al[3] Association Rule Mining (ARM) identifies frequent itemsets from databases and generates association rules by considering each item in equal value. However, items are actually different in many aspects in a number of real applications, such as retail marketing, network log, etc. The difference between items makes a strong impact on the decision making in these applications. Therefore, traditional ARM cannot meet the demands arising from these applications. By considering the different values of individual items as utilities, utility mining focuses on identifying the itemsets with high utilities. As "downward closure property" doesn't apply to utility mining, the generation of candidate itemsets is the most costly in terms of time and memory space. In this paper, we present a Two-Phase algorithm to efficiently prune down the number of candidates and can precisely obtain the complete set of high utility itemsets. In the first phase, we propose a model that applies the "transaction-weighted downward closure property" on the search space to expedite the identification of candidates. In the second phase, one extra database scan is performed to identify the high utility itemsets. We also parallelize our algorithm on shared memory multi-process architecture using Common Count Partitioned Database (CCPD) strategy.

Tsai, Pauray et al[4]Most of researches on mining high utility itemsets focus on the static transaction database, where all transactions are treated with the same importance and the database can be scanned more than once. With the emergence of new applications, data stream mining has become a significant research topic. In the data stream environment, online data stream mining algorithms are restricted to make only one pass over the data. However, present methods for mining high utility item sets still cannot meet the requirement. In this paper, we propose a single pass algorithm for high utility item set mining based on the weighted sliding window model. The developed algorithm takes advantage of reusing stored information to efficiently discover all the high utility itemsets in data streams.

Wu, Cheng Wei, et al [5] Mining high utility itemsets from databases is an emerging topic in data mining, which refers to the discovery of itemsets with utilities higher than a user-specified minimum utility threshold min_util. Although several studies have been carried out on this topic, setting an appropriate minimum utility threshold is a difficult problem for users. If min_util is set too low, too many high utility itemsets will be generated, which may cause the mining algorithms to become inefficient or even run out of memory. On the other hand, if min_util is set too high, no high utility itemset will be found. Setting appropriate minimum utility thresholds by trial and error is a tedious process for users. In this paper, we address this problem by proposing a new framework named top-k high utility itemset mining, where k is the desired number of high utility itemsets to be mined

V. Conclusion

Mining Cost Efficient item-sets from a transactional database refers to the discovery of transaction sets with Cost Efficient characteristics improving profits. Quite a few relevant algorithms have been proposed in the research community, however they bring upon the problem of producing a large number of candidate item-sets for Cost Efficient item-sets. Such a large number of candidate item-sets degrade the mining performance in terms of execution time and space requirement. Which can make situation even worse when databases contain very long transactions. In this paper, we propose new algorithm, for mining Cost Efficient item-sets with a set of effective strategies for pruning candidate item-sets. The information of Cost Efficient item-sets is maintained in a tree-based data structure such that candidate item-sets can be generated efficiently with only two scans of database. In future we would like to propose a method for mining efficient Cost Algorithms for reducing High Toll Transactions.

References

[1] Liu, Mengchi, Junfeng Qu.,"Mining high utility itemsets without candidate generation", In Proceedings of the 21st ACM international conference on Information and knowledge management, pp. 55-64.ACM, 2012.

[2] Samadi, Pedram, A-H. Mohsenian-Rad, Robert Schober, Vincent WS Wong, JuriJatskevich,"Optimal real-time pricing algorithm based on utility maximization for smart grid", In Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on, pp. 415-420. IEEE, 2010

[3] Liu, Ying, Wei-keng Liao, AlokChoudhary,"A fast high utility itemsets mining algorithm", In Proceedings of the 1st international workshop on Utility-based data mining, pp. 90-99.ACM, 2005.

[4] Tsai, Pauray SM,"MINING HIGH UTILITY ITEMSETS IN DATA STREAMS BASED ON THE WEIGHTED SLIDING WINDOWMODEL", International Journal (2014)

[5] Wu, Cheng Wei, Bai-En Shie, Vincent S. Tseng, Philip S. Yu.,"Mining Top-K high utility itemsets", In Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 78-86. ACM, 2012

[6] Hong, Tzung-Pei, Cho-Han Lee, and Shyue-Liang Wang. "Mining high average-utility itemsets", In Systems, Man and Cybernetics, 2009.SMC 2009. IEEE International Conference on, pp. 2526-2530. IEEE, 2009.

[7] Cardosa, Michael, Madhukar R. Korupolu, Aameek Singh,"Shares and utilities based power consolidation in virtualized server environments", In Integrated Network Management, 2009.IM'09. IFIP/IEEE International Symposium on, pp. 327-334. IEEE, 2009.

[8] Liu, Ying, Wei-keng Liao, AlokChoudhary,"A two-phase algorithm for fast discovery of high utility itemsets", In Advances in Knowledge Discovery and Data Mining, pp. 689-695. Springer Berlin Heidelberg, 2005.